Claims Amendments

Please replace the previous claims with the claims listed below.

Claims 1-67 (canceled).

68. (new) A method of digital signal processing of multi-sampled phase (DSP MSP), recovering data from a received signal, which comprises capturing multiple samples of the signal per a symbol time with a sampling clock or its sub-clocks defining known phase displacements versus the sampling clock; the DSP MSP method comprising the steps of:

detection of phases of rising and falling edges of the signal by using said signal samples captured at said known phase displacements;

evaluation of a length of a pulse of the signal by using said phases of signal edges;

calculation of a number of data bits received in the pulse by using said evaluation of the pulse length.

69. (new) A method as claimed in claim 68 further comprising a processing of said captured signal, wherein such captured signal processing comprises:

utilizing multiple sequential processing stages, driven by the sampling clock or clocks synchronous to the sampling clock;

utilizing multiple parallel processing phases, wherein consecutive parallel phases are driven by clocks which are shifted in time by one or more periods of said sampling clock;

passing outputs of a one parallel processing phase to a next parallel phase, wherein output register bits of the original parallel phase are re-timed by clocking them into an output register of the next parallel phase simultaneously with processing results of the next parallel phase;

using said passed outputs for processing conducted by a following sequential processing stage which belongs to the next parallel processing phase.

70. (new) A method of digital signal processing of multi-sampled phase (DSP MSP), recovering data from a received signal, which comprises capturing multiple samples of the signal per a symbol time with a sampling clock or its sub-clocks defining known phase displacements versus the sampling clock; the DSP MSP method comprising the steps of:

filtering out noise from said captured signal with digital filters;

detecting phases of rising and falling edges of the resulting filtered signal derived from said signal samples captured at said known phase displacements;

evaluation of a length of a pulse of the filtered signal by using said phases of filtered signal edges;

calculation of a number of data bits received in the pulse by using said evaluation of the pulse length.

71. (new) A method as claimed in claim 70, wherein said evaluation of pulse length comprises the steps of:

defining an edge skew, between an edge of the sampling clock and the signal edge, with the time displacement of sub-clock which captures a change in a signal level;

measuring said pulse length as being composed of such edge skew of a front edge of the incoming waveform, an integer number of sampling clock periods between the front edge and an end edge, and such edge skew of the end edge of the waveform.

72. (new) A method of digital signal processing of multi-sampled phase (DSP MSP) for recovering data from a received signal waveform, captured with a sampling clock or its sub-clocks, by processing length of inter-transition intervals of the captured waveform; the DSP MSP method comprising the steps of:

capturing multiple received signal levels during every symbol time by the sampling clock or its sub-clocks;

converting such captured data into transition times of the received signal waveform;

measuring said length of inter-transition interval occurring between said transition times;

calculating a number of data bits received during the inter-transition interval by evaluating said lengths of inter-transition interval.

73. (new) A method of synchronous sequential processing (SSP) for sampling and capturing and processing of a waveform, wherein said waveform sampling and capturing use a sampling clock or outputs of a sampling clock delay line and said waveform processing uses multiple sequential processing stages or multiple parallel processing phases; wherein the SSP method comprises the steps of:

driving said sequential processing stages with clocks synchronous to said sampling clock,

and performing a cumulative processing operation split into a series of consecutive basic operations implementing addition or subtraction or comparison wherein a result of one basic operation performed earlier is used for processing a result of another basic operation performed later;

or driving said parallel phases with clocks synchronous to said sampling clock wherein consecutive parallel phases are driven by clocks shifted in time by one or more periods of the sampling clock,

and passing outputs of a one parallel phase to a next parallel phase in order to use said passed outputs for processing conducted by a following stage of the next parallel phase wherein said outputs passing is performed by re-timing output register bits of the one phase by clocking them into an output register of the next parallel phase simultaneously with processing results of the next parallel phase;

wherein said use of earlier sequential operation result for processing later sequential operation result or said passing of outputs of one parallel phase to next parallel phase, enables continuous processing of indefinite waveform intervals carrying high frequency pulses.

74. (new) An SSP method as claimed in claim 73 comprising merging of said parallel processing phases, the SSP method comprising the step of:

merging said multiple parallel processing phases into a smaller number of parallel phases or into a single processing phase, when passing from a one sequential processing stage to a next sequential stage;

wherein clocking frequency of such merged phase equals to a sum of clocking frequencies of said parallel phases which are clocked into the merged phase.

75. (new) An SSP method as claimed in claim 73 further comprising synchronous sequential stages or parallel processing phases for noise filtering, the SSP method comprising the steps of:

clocking-in carry over bit or bits of an output register of a first filter stage of said one parallel phase into an output register of the first filter stage of the next phase together with filtering results of the next phase;

using the output register of the first filter stage of one phase by a second filter stage of next phase for filtering a wave-form interval which extends through both said parallel phases.

76. (new) An SSP method as claimed in claim 73, wherein:

said sequential processing stages use selectors or arithmometers or output registers.

77. (new) An SSP method as claimed in claim 76 wherein said waveform processing further comprises use of multiple parallel processing stages; wherein;
said multiple parallel processing stages, performing different logical or arithmetical operations, are driven by the same clock which is applied simultaneously to all the parallel stages.

78. (new) An SSP method as claimed in claim 73 further comprising use of a programmable control unit (PCU)) for implementing programmable or adaptive signal processing algorithms; the SSP method comprising the steps of:
using said waveform processing, utilizing sequential processing stages or parallel processing phases, for real time capturing and processing of an in-coming waveform;
using said PCU for reading results of the waveform processing from said synchronous sequential stages or parallel processing stages and for controlling operations of the waveform processing.

79. (new) An SSP method as claimed in claim 78, the SSP method further comprising:
screening and capturing of the incoming signal with a wave-form screening and capturing circuits (WFSC) controlled by the PCU.

80. (new) An SSP method as claimed in claim 79, wherein the SSP method further comprises:
using said WFSC for verification of said captured waveforms for compliance or non-compliance with programmable patterns and for buffering captured waveform for which the preprogrammed compliance or said non-compliance has been detected;
wherein said programmable patterns are provided by the PCU and such buffered waveform is read by the PCU.

81. (new) An SSP method as claimed in claim 79, wherein operations of said WFSC further comprise:
selecting a time interval for which incoming wave-form captures shall be buffered and communicated to the PCU, by programming a time slot selection circuit;
wherein such slot selection is programmed by the PCU and such pre-selected buffered waveform is read by the PCU.

82. (new) An SSP method as claimed in claim 73 comprising use of a periodical skew accumulation (PSA) circuit for correcting cumulative error caused by periodical phase skews; the SSP method comprising the steps of:

using a periodical skew as an estimate of a phase skew between the sampling clock period versus an expected period of a clock which drives the incoming signal;

using the PSA for calculating an accumulation of said periodical skews for a single pulse or for a combinations of pulses of the incoming signal;

using such periodical skew accumulation to correct a length of a single data string or multiple data strings detected.

83. (new) An SSP method as claimed in claim 82; wherein operations of said PSA comprise the steps of:

reading a next set of said periodical skews from the PCU or other circuits and attaching them to a present set of the periodical phase skews;

synchronous communication of said accumulations of the pulse skews to phase processing stages which use such accumulations to modify a lengths of said single data string or multiple data strings detected.

84. (new) A method of fractional bit staffing (FBS) for improving accuracy of fixed point arithmetic over that of conventional solutions for a long cumulative processing operation split into a series of basic addition or subtraction or comparison operations between components into a series of basic addition or subtraction or comparison operations on or between components of a processed argument and terms of a processing argument; the FBS method comprising the steps of:

expressing said processing argument as a series of terms, wherein each term may have a differently staffed last bit or several last bits expressing a fractional value of the term;

combining said staffed last bits with previous bits expressing more significant constant part of the term in order to provide said term;

a series of such terms is provided for a repeatedly performed arithmetic operation;

using every consecutive term, of a processing argument of said cumulative operation, for processing performed during a corresponding consecutive basic operation;

wherein the FBS enables reduction of a total error of such long cumulative operation to a single last bit resolution;

85. (new) An FBS method as claimed in claim 84, wherein:
said series of multiple binary terms is downloaded to a register;
a circuit which performs said arithmetic operations uses said terms
by shifting the register during any operation and accessing the same portion of the register
or by utilizing a selector circuit which selects a consecutive portion of the register which contains the corresponding term.

86. (new) An FBS method as claimed in claim 84, comprising correction of cumulative errors in measuring lengths of incoming signal inter-transition interval with a local sampling clock having periodical phase skews versus symbol periods expected in an incoming signal; wherein:
a series of said periodical phase skews is derived as estimates of a phase skew between a sampling clock period versus the expected symbol period anticipated for consecutive symbol periods sampled during the inter-transition interval, wherein such phase skew estimates may differ for said consecutive symbol periods;
the series of periodical phase skews, used as said differently staffed last bits expressing said fractional values, are combined with a sampling clock period, used as the more significant constant part of the terms, in order to produce said terms.

87. (new) A clock selection system (CSS) for enabling sub-clocks, generated by the outputs of serially connected gates which a sampling clock is propagated through, during particular phases corresponding to cycles of the sampling clock; wherein:
clock selectors connected serially are used for enabling said sub-clocks;
falling edges of said sub-clocks are used for driving said clock selectors selecting parallel processing phases during which positive sub-clocks are enabled;
or rising edges of said sub-clocks are used for driving said clock selectors selecting parallel processing phases during which negative sub-clocks are enabled.